

Kairos Chat: 主観的時間の概念を導入したチャットシステム

小倉 加奈代[†] 松本 遥子[†] 山内 賢幸[†] 西本 一志[†]

本稿では、流速が異なる複数の時間流を持つチャットシステム“Kairos Chat”を提案する。対面口頭での対話では、人は議論の本筋とは関係の無い逸脱発言を行うことにより、議論の円滑化を図っている。同時に、逸脱発言を急速に忘却することで、議論記憶を自然に精練化している。これは、人が各発言を、その内容に応じて異なる主観的時間流上で扱っている結果と見ることができる。しかし、従来のチャットシステムでは、すべての発言を単一の時間流上で扱うため、議論記憶としての発言履歴上に逸脱発言が混在し、精練化が行われない。そこで Kairos Chat では、各発言を内容に応じて異なる流速の時間流上で扱うことを可能とすることにより、逸脱発言のしやすさと、自然な発言履歴の精練化の両立を目指す。被験者実験の結果、ユーザは自然に流速の違いを活用し、逸脱発言を気軽に行うとともに、発言履歴の精練化が部分的に実現されることがわかった。また、各時間流と対話スレッドの関係性についても調査し、対話スレッドの中の4割の隣接ペアが異なる時間流をまたぐものであることも明らかとなった。

Kairos Chat: A Novel Text-Based Chat System That Innovates Subjective Time

KANAYO OGURA[†] YOKO MATSUMOTO[†] YOSHIYUKI YAMAUCHI[†] KAZUSHI NISHIMOTO[†]

In this paper, we propose a novel chat system named "Kairos Chat" that has multiple streams of time whose velocities are different. In a face-to-face communication, people facilitate a discussion using digressions and, at the same time, they naturally organize memories of the discussion by quickly forgetting them. We can regard this phenomenon as that they handle each utterance on a different stream of time based on its contents. However, the ordinary chat systems handle all of the utterances on a single stream of time. Therefore, the digressions are mixed into a chat log and the log cannot be naturally organized. Kairos Chat allows the users to handle each utterance on the different stream of time so as to strike a good balance between easy digressing and natural organizing of the chat log. From experimental results with subjects, we found that the subjects naturally utilize the different streams of time, they readily digress and the natural organizing of the chat log can be partially achieved. We also investigated the relations between the streams of time and conversation threads and found that half of the threads drift among the streams.

1. はじめに

比較的リラックスした会議などでは、議論の本筋とは直接関係ない、単純な言葉の意味の確認質問や軽い冗談などの「逸脱発言」も多数なされる。適切・適度な逸脱発言には、共通基盤形成の円滑化や、会議の雰囲気や和らげ議論を活性化する効果がある。このため、逸脱発言を排除するのではなく、むしろ会議の中でうまく活用することが望ましい。

しかし、逸脱発言の内容は、本質的な議論と無関係な場合も多いため、たとえば議事録に逸脱発言もすべて記録すると、議論の本筋を理解する妨げとなる。対面口頭での会議では、逸脱発言がなされた瞬間にはそれを受容して活用しつつ、一方でそれを急速に記憶の表層から消し去ることで、議論に関する記憶を常時自然に精練している。これは、人が各発言を単一の客観的時間流（クロノス時間）上で扱うのではなく、経過

速度が異なる複数の主観的時間流（カイロス時間）上で扱い、発言の内容に応じて急速に忘れたり長く記憶にとどめたりしている結果であると見ることができる。

近年、計算機を介したコミュニケーションメディア（Computer Mediated Communication Media: CMC メディア）が多く利用されている。その一種であるテキストチャットやインスタント・メッセージング・システムなどの、テキスト情報をほぼリアルタイムでやり取りするメディア（以下、このようなメディアを総称して「テキストチャットメディア」と呼ぶ）は、その簡便性や口頭対話に近い使用感覚のために、簡易な遠隔会議システムなどとして広く利用されている。

従来のテキストチャットメディアは、発言をすべて単一の客観的時間流上で扱っており、多くのシステムがすべての発言を発言順に記録して表示する「発言履歴」を有する。これにより、任意の過去の発言を随時読み返すことができるため、複数の話題を同時進行させる「マルチスレッド対話」[1]のような、対面口頭対話では不可能な新たな対話形態を実現できるという

[†] 北陸先端科学技術大学院大学

利点を有する。しかし一方で、発言履歴には本質的発言も逸脱発言も渾然一体となって並んでいるため、その議論に参加していた場合ですら、発言履歴を読んで議論の流れを正確に把握することが難しい。このため、逆にユーザが逸脱発言を控えてしまい、議論の円滑化や共通基盤形成が十分なされないケースも生じうる。

そこで本研究では、テキストチャットメディアに主観的時間流の概念を導入し、発言の内容に応じて各発言のエージング速度を変えることを可能とした、新たなチャットシステム“Kairos Chat”を提案する。Kairos Chat を用いれば、主観的時間の流れの違いに基づく議論記憶の精練化と類似した状態を発言履歴上に実現できると期待される。同時に、逸脱発言が発言履歴上における議論の本筋を断ち切る懸念がなくなるので、より柔軟かつタイムリーに気兼ねなく逸脱発言を行えるようになることも期待できる。

以下、第2章では関連研究について概観する。第3章では Kairos Chat のシステム構成について述べ、第4章では Kairos Chat の有用性を評価するための被験者実験について述べる。第5章では、実験結果について述べ、第6章では、実験結果に基づき Kairos Chat の有効性と特徴を考察する。第7章は、まとめである。

2. 関連研究

本章では、動的な発言履歴を有するチャットシステム、発言履歴の精練化が可能なチャットシステムの2種を取り上げ、Kairos Chat との比較を行う。

2.1 動的な発言履歴を有するチャットシステム

動的な発言履歴を有するチャットシステムとして、Fugue[2]、Alternative Interfaces for Chat[3]を取り上げる。どちらのシステムも、発言履歴の横軸を時間軸とし、発言入力状況を発言履歴に反映させ、可視化することで、発言タイミングの取りにくさを解決するために開発されたシステムである。Fugue では、文字入力情報を逐次発言履歴に反映させるが、Alternative Interfaces for Chat では、一発言単位の入力情報を逐次発言履歴に反映させるタイプのチャットシステムであるという違いがある。両者とも、動的な発言履歴を有する点で、本提案システム Kairos Chat と共通するが、時間流は一般的チャットと同様に単一であり、議論記憶の精練や逸脱発言の発言しやすさを目的とはしていない点でも本提案システムと異なる。

また、一般的なチャットシステムと異なるが、KJ法支援システム上で、チャットを用いたブレインストーミングの結果から有用な発言を拾い出す作業のため

に、時間軸を取り入れた「発言が流れる」インタフェースを提案した研究[4]がある。この研究でインタフェース上に存在する流れは、カードのシャッフルと同等の効果を狙ったものであり、すべての発言がやはり単一の時間流で扱われている点で本研究とは異なる。

2.2 発言履歴の精練化が可能なチャットシステム

発言履歴の精練化が可能なチャットシステムとして、遠隔ゼミナール支援システム RemoteWadaman V [5]のセマンティック・チャットがある。このシステムでは、ユーザは発言を入力した後、個々の発言に対し、「Idea」(着想, 意見, 提言), 「質問」, 「返答」, 「感想」, 「メモ」, 「あいさつ」の合計9種類のタグのいずれかを発言意図に応じて明示的に付与することが求められる。セマンティック・チャットでは、この付与されたタグに応じた発言履歴の精練化が可能である。これに対し Kairos Chat では、ユーザがどの時間流へ発言を投入するかにより、発言タイプを粗く区別することができ、それと同時に発言履歴の精練化が可能である。セマンティック・チャットでは、あらかじめ用意されたタグのいずれかを発言に付与する必要があるが、本提案システムでは、各発言に対して意図的に明示的なタグ付けを要求しているわけではなく、ユーザの自然で暗黙的な発言行動にまかせることで、発言履歴の精練化が可能となり、この点で両者は大きく異なる。

3. システム概要

今回開発したチャットシステムは、Web アプリケーションとして実装した。サーバは、Microsoft Windows XP を使用し、クライアント側の処理を Adobe Flash で、サーバ側の処理を php で行っている。

3.1 サーバ概要

サーバは、クライアントから受信した発言データ(名前, 発言内容, 日時, レーン(投入先の時間流に対応))を保存するモジュールと、クライアントから最新のログを要求された時に受け渡すモジュールの2つのモジュールから成る。発言を保存するモジュールは、発言を受け取ったら、その発言がどのレーンに投入されたものかに応じて、各レーンに対応する保存場所に発言の受け取り順に保存する。最新ログをクライアントに受け渡すモジュールは、クライアントが保持している最新の発言番号をサーバ上のログの最新発言番号と比較し、差分だけをクライアントに返す。

3.2 クライアント概要

クライアントのシステムは、Adobe Flash で作成しており、ユーザは Web ブラウザ上で実行する。

Firefox 3.0.10 で動作の確認を行った。図 1 に、Kairos Chat のユーザインタフェースを示す。上部には名前とメッセージを入力するテキストボックス、下部にはログが表示される 3 つの発言履歴表示レーン（以下単に「レーン」とする）が配されている。最も左側のレーンは、メッセージが上から下まで 8 秒で流れる「Fast」レーン、中央のレーンはメッセージが上から下まで 40 秒で流れる「Slow」レーン、最も右側のレーンは、通常のチャットと同様、新しい発言が最上部に追加されると、古いものは順に下へ押し出されていく「Push」レーンである。

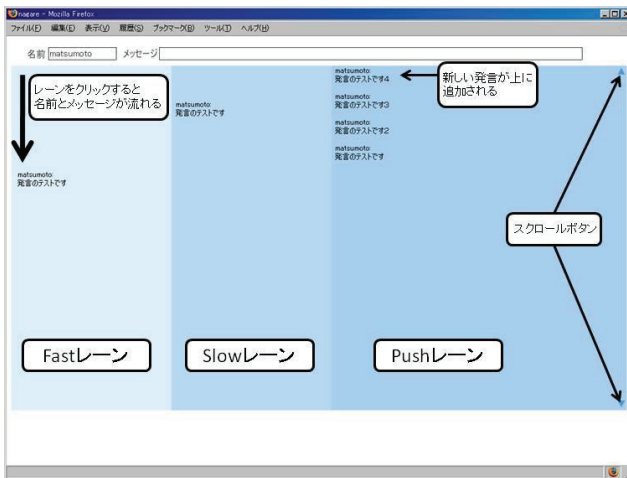


図1 Kairos Chat クライアントの UI

ユーザは、メッセージ入力欄にメッセージを入力し、これら 3 つのレーンのうち、当該メッセージを流したいレーンをクリックする。すると、クリックしたレーンの上部に入力したメッセージが投入され、Fast レーンと Slow レーンでは時間経過と共にメッセージが上から下へ流れ落ちる。なお、Push レーンのみにはスクロールボタンが用意されており、下スクロールボタンにマウスカーソルを乗せることによって過去の発言履歴を閲覧できる。Fast レーンと Slow レーンについては、過去の発言を見返す機能は提供していない。各レーンのリロード時間は、Fast・Push レーンは 2 秒に 1 回、Slow レーンは 4 秒に 1 回である。これは、リロード時間内に複数の投稿があった場合、文字が重なってメッセージが読めなくなることを防ぐためである。

4. 実験

本章では、「はじめに」で示した、Kairos Chat による自然な発言履歴の精錬化と、逸脱発言のしやすさという 2 つの効能を確認するために、以下の仮説を検証するための実験を行う。

- A) ユーザは、特別な教示なしに、自発的に各レーンを発言内容に応じて使い分ける。
- B) Push レーンには議論の本筋となる発言が主に投入され、議事録的な発言ログが形成される。
- C) その他のレーン（特に Fast レーン）の存在により逸脱発言がしやすくなる。

さらに、Kairos Chat を用いた際の対話スレッド構造と各レーンの関係についても調査する。そのために、実験では、以下の 3 つの工程を実施した。

- 1) システム利用実験
- 2) 提案システム利用時の発言タイプ評価
- 3) システム利用に関するアンケート調査

次節より上記の各工程について説明する。

4.1 システム利用実験

4 人の大学院生からなる被験者群 7 組、計 28 人に對し、以下の 2 つのシステムを用いた実験を行った。

- **Baseline Chat:** 提案システム Kairos Chat の右側「Push」レーンのみを持つチャット（図 2）。発言送信方法が Kairos Chat と同じである点（レーンをクリックして送信）以外は、一般的チャットシステムと同じ機能を有する。
- **Kairos Chat:** 前章で説明した本研究提案システム

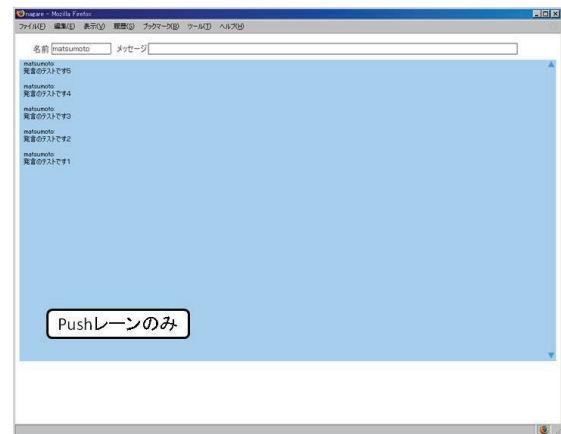


図2 BaselineChat の UI

各被験者群に対し、セッション 1 : Baseline Chat → セッション 2 : Kairos Chat → セッション 3 : Baseline Chat の 3 セッション（1 セッション約 30 分）の実験を行った。これは、提案システム Kairos Chat 使用後の Baseline Chat の使用感も調査するためである。

チャットの課題は、協調的意思決定課題として以下 3 つの課題を、各被験者群に順番を変え適用した。これは、課題が対話に対して及ぼす影響を最小にするための配慮である。

- 1) 研究室で合宿に行くとしたらどこで何を

か？

- 2) このメンバーで合コンをしたらどこでどのように行うか？
- 3) 指導教員へ誕生日プレゼントを贈るとしたら何を贈るか？

被験者群の構成は、同じ研究室もしくは同一学年メンバーで構成され、互いに面識がある関係であり、事前にチャットの相手が誰かを知らされている。なお、視覚や声による意思疎通を排除するため、被験者は全員離れた個室で実験を行った。また、被験者は全員、何らかの形でテキストチャットを使用したことがあり、日頃からキーボードを利用する環境に置かれている。そのため、発言入力に特に長時間を要する被験者はいなかった。

システムの利用方法については、実験開始前に、Baseline Chat, Kairos Chat とともに基本的な投稿方法と、Push レーンでの履歴閲覧方法のみを、全被験者へ教示した。各レーンにどのような発言を流すべきか、などの指示は一切行っていない。また、実験中は Chat 以外の、ブラウザの閲覧など他の操作は禁止した。

4.2 提案システム利用時の発言タイプ評価

Kairos Chat の 3 つのレーンと発言内容との間に使用傾向の違いがあるのかを調べるため、全被験者に対し、Kairos Chat を用いた対話でなされた全発言について、議論との関連度合いによって設定した 8 つの発言タイプのいずれに該当するかを主観的に評価してもらった。なお、設定した 8 つの発言タイプについては、5.3 節にて説明する。

4.3 システム利用に関するアンケート調査

セッション 1 と 2 の終了後、使用したチャットシステムについてアンケート調査を行った。なお、調査したアンケート項目については、5.6 節にて説明する。また、Kairos Chat の利用セッション後のみ、発言履歴を印刷したものを提示し、全被験者に対し、見覚えのない発言をチェックしてもらった。これは、動的なレーンでの読み逃しが無いかを調べるためである。

5. 実験結果

本章では、5.1 節および 5.2 節で、システム利用実験時の各セッション間の発言数の比較結果と、提案システム Kairos Chat における各レーンの発言数を示す。次いで、5.3 節では、4.2 節で述べた Kairos Chat 使用時の発言タイプについての結果について述べる。5.4 節および 5.5 節では、Kairos Chat を用いた時の対話ス

レッド構造の特徴を明らかにするために、各セッション間の話題（スレッド）数を比較し、Kairos Chat における各レーンとスレッドとの関係を調査する。最後に 5.6 節では、4.3 節で述べたアンケート結果、および見覚えのない発言の調査結果について述べる。

5.1 各セッションでの発言数

Baseline Chat と Kairos Chat で発言頻度に差があるかを見るため、各セッションでの発言数と、単位時間（秒）あたりの発言数を調査した（表 1）。なお、数値は各被験者群の平均である。

表1 各セッションでの発言数と単位時間あたりの発言数

	S1	S2	S3
発言数	129.1	165.7	140.9
発言数/時間(秒)	0.071	* 0.087	0.075

N=7, *: $p < .05$

表 1 の発言数の比較から、S2 の Kairos Chat を用いた場合に発言数が増えていることがわかる。ただし、各セッションで対話時間にばらつきがある（S1:1813 ± 45sec., S2:1918 ± 60sec., S3:1904 ± 41sec.）ので 1 秒あたりの発言数で比較すると、Baseline 使用後に Kairos Chat を使用した際に発言数の有意な増加が認められた。

5.2 Kairos Chat 利用時の各レーンの発言数

Kairos Chat について Fast レーン, Slow レーン, Push レーンの 3 つのレーンについての使用頻度を見るために、それぞれのレーンの発言数を調査した（表 2）。なお、数値は各被験者群の平均値である。

表2 Kairos Chat 利用時の各レーンの発言数

Fast	Slow	Push
50.4	73.4	* 41.9

N=7, *: $p < .05$

表 2 より、Slow レーンのほうが、Push レーンよりも有意に発言数が多いことが認められた。

5.3 提案システム利用時の発言タイプ評価

4.2 節で述べたように、Kairos Chat の 3 つのレーンと発言内容との間に使用傾向の違いがあるのかを調べるため、全被験者に対し、Kairos Chat を用いた対話でなされた全発言について、以下の 8 つのタイプのいずれに該当するかを主観的に評価してもらった。

- (1) 議題と密接に関連した公式発言（関連公式）（会議中に挙手が必要な類の発言）
- (2) 議題と密接に関連した非公式発言（関連非公式）（会議中の独り言、隣人との一時的対話、突発的発言などに類する発言）

- (3) 議題と関連がある周辺的な話題に関する発言 (関連周辺) (単純な語句の意味の確認など)
- (4) 議題とあまり関係がない発言 (弱関連)
- (5) 議題と全く関連ない話題に関する発言 (無関連)
- (6) 冗談
- (7) あいづち
- (8) その他

上の8つの発言タイプの評価後、レーン毎に各タイプの発言がいくつ含まれていたかを数えた。被験者が自分自身の発言のみに評価した結果(送り手側の評価)から求めた、各発言タイプにおける各レーンの使用割合を図3に示す。また、被験者が自分以外の発言に評価した結果(受け手側の評価)から求めた、各発言タイプにおける各レーンの使用割合を図4に示す。

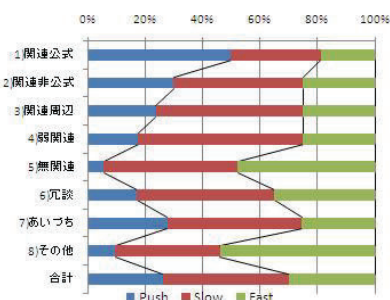


図3 送り手側からの発言タイプとレーンの関係評価(割合)

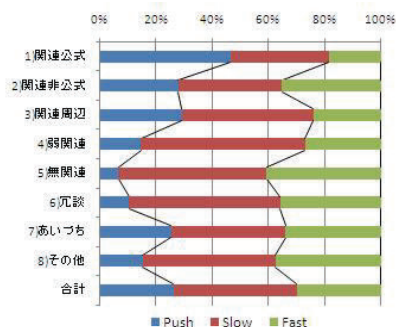


図4 受け手側からの発言タイプとレーンの関係評価(割合)

図3,4の結果から、各レーンの発言タイプの傾向をみると、以下のことがわかる。

- Push レーンは、主として関連公式タイプに用いられ、関連非公式タイプとあいづちにも比較的多く用いられる。それ以外の、議題との関連が弱いタイプではあまり用いられない。
- Slow レーンは、関連公式発言以外のすべてのタイプで多用される。
- Fast レーンは、関連性が強いタイプではあまり用いられず、無関連、冗談、その他の、議題と

関係が無い話題に関するタイプで多用される。

また、各レーンの発言タイプの傾向を、送り手側と受け手側の評価した発言タイプに分けてみると以下のことがわかる。

- Push レーンについては、送り手側と受け手側の評価に顕著な差がある項目はなかった。
- Slow レーンについては、送り手側は受け手側よりも関連非公式、関連周辺の、議題とやや関係あるタイプが多いと評価し、受け手側は送り手側よりも無関連、冗談、その他の、議題と関係がないタイプが多いと評価している。
- Fast レーンについては、送り手側は受け手側よりもその他や無関連タイプをやや多く評価し、受け手側は関連非公式とあいづちタイプをやや多く評価している。

5.4 スレッド数の比較

1章で述べたように、テキストチャットメディアは、複数の話題を同時進行させる「マルチスレッド対話」を可能とすることが特徴の1つである。テキストチャットメディアにおける、1つの話題に関する発言の連鎖(以下スレッドと呼ぶ)の推移は、対面口頭対話よりもめまぐるしく、インタフェースや、操作性の違いにより変化しやすい。そこで本研究でも、スレッドの推移に着目し、Baseline Chat と Kairos Chat の間で生じるスレッド数の比較を行った。

なお、スレッド数の算出に先立ち、各セッションの全対話データに対し、スレッド同定作業を行った。この作業は、著者らが考案した方法[6]をもとに、信頼性を高めるため、2名の作業者で行った。この作業後に、各セッションでのスレッド数と、1スレッド内の発言数(発言数/スレッド数)を算出した(表3)。

表3 各セッションでのスレッド数とスレッド数/発言数

	S1	S2	S3
スレッド数	12.1	24.9	10.1
発言数/スレッド数	12.3	7.0	14.8

N=7, *: $p < .05$, **: $p < .01$

表3のスレッド数の比較からは、S2のKairos Chatを用いた場合に有意にスレッド数が増えていることが認められる。また、1スレッド内の発言数は、Kairos Chatのほうが、Baseline Chatよりも有意に少ないことがわかる。この結果から、Kairos Chatでは、Baseline Chatよりも1スレッドが短く、スレッドが頻繁に推移していることがわかった。

5.5 Kairos Chat での各レーンとスレッドの関係

Kairos Chat は、時間流の異なる3つのレーンを有

する。そこで、あるスレッドが進行する場合、1 レーンの中でスレッドが進行するのか (図 5 左)、あるいは複数のレーンを行き来しながら進行するのか (図 5 右) が興味の対象となる。特に複数レーンを行き来するスレッドが多発するならば、発言対の関係がわかりにくくなるため、将来的になんらかの UI 上の対策が必要となるかもしれない。そこで、レーンとスレッド中での発言の推移の関係を調査した。

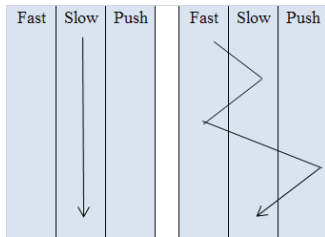


図5 レーンとスレッドの関係例

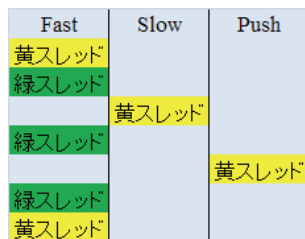


図6 またぎ度数説明図

調査方法について、図 6 を用いて説明する。図 6 にて、色のついたセルは 1 発言である。また、黄色、緑色は異なるスレッドを表す。各スレッド内の発言の推移状態を見ると、黄では、Fast→Slow→Push→Fast と推移し、緑では、Fast→Fast→Fast と推移している。この推移状態について、異なるレーンに遷移した場合を「1」、同一レーン内で続いた場合を「0」とカウントすることとする。よって図 6 では、黄スレッドは「3」、緑スレッドは「0」となる。こうして求めた得点を発言推移数 (= 各スレッド内の発言数 - 1) で割ったものを「遷移割合」と呼ぶことにする。

図 6 の例に戻ると、黄スレッドの遷移割合は $3/3 = 1.0$ となり、緑スレッドの遷移割合は $0/2 = 0.0$ となる。遷移割合が 0.0 の場合は直線推移、1.0 の場合は、全推移がレーンをまたいだ状態であることを表す。

Kairos Chat を用いた各発言履歴について、開始発言から順番に、スレッドを考慮せずにレーンに対する推移を追った場合の遷移割合と、図 6 のように、スレッドを考慮した遷移割合を算出した (表 4)。なお、数値は各被験者群の平均値である。

表 4 より、スレッドを考慮しない場合、推移状態が直線的になる部分と複数レーンをまたいだ部分が半々

であることがわかる。また、各々のスレッドでみた場合、直線的な推移の方が若干多いが、複数レーンをまたいで推移も 4 割弱起っていることがわかる。図 7 に、1 被験者群における Kairos Chat を用いた場合のスレッド推移図を示す。このように、比較的直線的に同一レーン内で進行するスレッドと、きわめて頻繁にレーンをまたぐスレッドとに分かれる様子が見て取れる。

表4 スレッド考慮なしとありの遷移割合

	スレッド考慮なし	スレッド考慮あり
遷移割合	0.51	0.38

N=7, **: $p < .01$

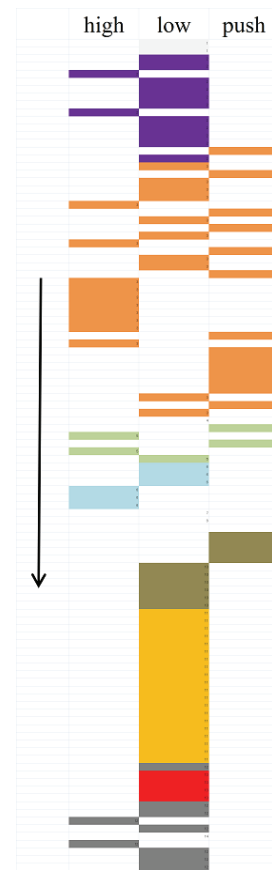


図7 一被験者群のスレッド推移図 (同一色が 1 スレッド)

5.6 システム利用に関するアンケート調査

セッション 1 と 2 の終了後に実施したアンケートの調査項目と結果を表 5 に示す。評価は、各項目 5 段階で、質問項目にあてはまる場合「5」、あてはまらない場合を「1」として設定した。No.に振った*および**は、それぞれ当該質問項目における評価結果に 5%水準あるいは 1%水準で有意差があったことを示す。表 6 には、セッション 2 終了後に調査した、Kairos Chat の各レーンの使用感に関するアンケート項目と結果を

示す。評価は、あてはまるレーンを1つ選択する方式で行った。

表5 セッション1と2終了後アンケートとその結果

No.	質問内容	Kairos	Base
1*	このチャットシステムは使いやすかったですか	3.29	2.79
2	このチャットシステムは直感的に操作できましたか	3.52	3.30
3	操作にストレスを感じずに使うことができましたか	3.21	2.96
4	レーンの選択はスムーズにできましたか	3.50	---
5**	このチャットシステムは発言がしやすかったですか	3.96	3.18
6	このチャットシステムは議論がまとめやすかったですか	2.93	2.75
7	このチャットシステムは議論がスムーズに進みましたか	3.11	2.93
8	テーマと関係する発言を発言しやすかったですか	3.36	3.29
9**	テーマと直接関係ない単純な質問をしやすかったですか	4.14	3.46
10**	冗談などのテーマと無関係な発言をしやすかったですか	4.18	3.29
11	議論の流れを追いやすかったですか	2.68	3.04
12	各レーンごとに発言内容を変えよう意識しましたか	3.82	---
13**	このチャットは面白かったですか	4.00	3.04
14**	今後も使い続けてみたいですか	3.29	2.5

N=28, *: $p < .05$, **: $p < .01$

注) 質問4, 12はKairos Chatのみに該当する項目である。

表6 レーンの使用感に関するアンケート調査結果

No.	質問内容	P	S	F
15	どのレーンが一番テーマと関係する発言がしやすかったですか	15	13	0
16	どのレーンが一番テーマと直接関係しない単純な発言を発言しやすかったですか	0	9	19
17	どのレーンが一番冗談などのテーマと無関係な発言をしやすかったですか	0	4	24

P:Push レーン, S:Slow レーン, F:Fast レーン

表5より、Kairos Chatのほうが使いやすいと評価されており(質問1)、発言のしやすさも高く評価されている(質問5)。また、発言のしやすさについては、テーマとは直接関係のない単純な質問と冗談に対して高く評価されている(質問9, 10)。さらに、Kairos Chatを使ってみておもしろいと評価されている(質問13)、Baselineよりは今後も使い続けてみたいと評価されている(質問14)。

また、Kairos Chatの複数レーンの使用については、表5から、特殊なインタフェースを持ちつつも、レーン選択で特に問題はなく(質問4)、操作性には問題

はなかったと評価されている。また、レーンごとに発言内容を変えようとする意識が働いていることが読み取れる(質問12)。この点について、表6の結果から、レーンによって発言しやすい発言タイプが異なることが示されている。テーマに関係する発言はFastレーン以外のどちらかで行い(質問15)、テーマと直接関係無い発言や冗談などの無関係な発言はFastレーンで発言しやすいと感じられている(質問16, 17)。

また表7に、記憶に残っていなかった発言の1人あたりの平均数を示す。この結果、記憶に残っていなかった発言は全体のわずか3%ほどであり、HighやSlowなどの動的なレーンでも記憶に残っていない発言数は少なく、読み逃しはほとんどなかったことがわかった。

表7 レーン毎の記憶に残っていなかった平均発言数

Fast	Slow	Push	総発言数
3.75	1.11	0.14	4471

6. 考察

6.1 仮説A:各レーンの自発的使い分け

5.3節と表6の設問12に示した結果より、ユーザは特に何の教示も受けなかったにもかかわらず、流速の違いを自然に受け入れ、レーンごとに発言内容を意識した使い分けを行っていることが明らかとなった。以上の結果から、仮説Aは支持されたとと言える。

6.2 仮説B:Pushレーンでの発言ログ精練

5.3節と表6に示した結果から、議題と密接に関連する発言はPushレーン上でもっとも多くなされることがわかった。しかし一方で、Slowレーン上でも議題との関連度が強い発言が多くなされることも明らかとなり、表5の設問6, 8, 11の結果に見られるように、議論のまとめやすさや議題と密接に関連する発言のしやすさについて有意差を見いだすことはできなかった。ただし、アンケートの自由記述に「重要度が低い順から発言した」、「早いレーン:ジョーク, 遅いレーン:議論, 止レーン:まとめ」、「真ん中で議論して、右に確定事項を並べるとか…」という記述があったことから、Pushレーンは、議論のまとめに使えるレーンとして認識されているということが示唆される。以上の結果から、仮説Bについては部分的に支持されたが、Pushレーンだけで議論の本筋をすべて把握可能かどうかについては不明であり、Slowレーン上の発言も加味するなど、さらなる検討が必要であることがわかった。

なお、5.3節で示したように、発言の送り手と受け

手の間で、特に Slow レーンと Fast レーン上の発言に対する受け止め方（発言タイプの判定）に差違が見られたことは興味深い。現在のシステムでは、発言の送り手の主観によってレーンが選択されているが、読み手側の主観によってレーン設定を変更できる機能などを追加することで、発言ログ精練化の個人適応を実現できる可能性がある。

6.3 仮説 C：逸脱発言のしやすさ

5.3 節と表 6 に示した結果から、議論との関連度が低い発言は Slow レーンや Fast レーン上で多くなされることがわかり、かつ表 5 の設問 9 と 10 の結果から、これら 2 つのレーンを備える Kairos Chat の方が、有意に逸脱発言をしやすかった。また、5.1 節で示した結果から、Kairos Chat の方が発言数は増える傾向にあり、表 5 の設問 5, 13, 14 で Kairos Chat の方が Push レーンしか持たない Baseline よりも、操作が複雑であるにもかかわらず、発言しやすく、面白く、より使いたいものであったことが示されている。以上の結果から、仮説 C は支持されたと言える。

6.4 Kairos Chat における対話の特徴

5.4 節に示したように、Kairos Chat では Baseline Chat よりもスレッド数が多く、かつ 1 スレッドに含まれる発言数が少ない。しかし一方で、5.1 節に示したように単位時間あたりの発言数は Kairos Chat の方が多い。つまり、Kairos Chat では 1 スレッドが短命であり、頻繁に新しいスレッドが生成される傾向がある。

また、5.5 節に示したように、Kairos Chat では、1 つのスレッドが単一のレーン上で形成されるケースもあるが、平均して 1 つのスレッド内に複数のレーンにまたがる発言が 4 割弱あることがわかった。5.3 節に示した発言タイプとレーンの対応を合わせて考慮すると、Push レーンや Slow レーン上の議題との関連が強い発言に対して、周辺的な質問や冗談などが Slow レーンや Fast レーンなどのより速い流速のレーンで行われるような使い方がなされているものと推測される。

このような実例として特に興味深かったのは、Fast レーンを使った Slow レーン上のコメントへの「追い越し応答」である。Slow レーンに投入されたコメントは、ゆっくりと下方に流れていく。このようなコメントに対して何か応答したい場合、応答コメントを Slow レーンに投入すると、両発言の間隔が開いたままとなって対応関係がわかりづらくなる。そこで、応答コメントを Fast レーンに投入し、Slow レーン上の対応コメントを追い越すことで、対応関係をわかりや

すくしようとしているものと思われる。また、追い越し応答コメントには、コメント文の最後に「→」を付加することで Slow レーンへの応答であることを明示することも多く見られた。このような使い方が、複数レーンをまたぐスレッドを形成したものと思われる。

7. おわりに

本稿では、経過速度が異なる複数の時間流を導入し、発言の内容に応じて発言のエイジング速度を変えることを可能とした新たなチャットシステム“Kairos Chat”を提案した。被験者実験により、ユーザは自発的に時間流を使い分けること、議事録的発言ログをある程度形成できること、逸脱発言がしやすくなることがわかり、全体としては Kairos Chat の有用性が示された。今後は、Push レーンと Slow レーンを中心としたレーン構成を見直すとともに、受け手側の主観を取り入れ可能とすることなどにより、的確な議事録的発言ログを生成可能とする手段を考案する予定である。

謝辞 本研究の一部は、(財)三谷研究開発支援財団平成 20 年度支援研究の助成を受けて実施された。また、実験データ分析のためのデータ処理作業で多大な協力をいただいた北陸先端科学技術大学知識科学研究科博士前期課程の金屋陽介氏に御礼申し上げる。

参考文献

- 1) 小倉加奈代, 西本一志: ChaTEL: マルチスレッド対話を容易にする音声コミュニケーションメディア, 情報処理学会論文誌, Vol.47, No.1, pp.98-111, 2006.
- 2) Rosenberger, Tara M., and Smith, Brian K.: Fugue: A Conversational Interface that Supports Turn-Taking Coordination, Proc. of HICSS2000, Vol.3, pp.3035, 2000.
- 3) Vronay, D., Smith, M. and Drucker, S.: Alternative Interface for Chat, Proc. Of the 12th Annual ACM Symposium on UIST, pp. 16-29., 1999.
- 4) Yuizono, T., Kayano, A. and Munemori, J.: Data Selection Interfaces for Knowledge Creative Groupware Using Chat Data, Proc. of KES2007, Part3, pp.446-452, 2007.
- 5) 由井蘭隆也, 重信智宏, 榎野晶文, 宗森 純: リアルタイムなコミュニケーション行為であるチャットへの意味タグ付加と電子ゼミナールへの適用, 情報処理学会論文誌, 47(1), pp.161-171, 2006.
- 6) Ogura, K., Ishizaki, M. and Nishimoto, K.: A Method of Extracting Topic Threads Towards Facilitating Knowledge Creation in Chat Conversations, Proc. of KES 2004, Part1, pp.330-336, 2004.